Package ‘epimdr’

May 8, 2026

Title Functions and Data for ``Epidemics: Models and Data in R''

Version 0.6-5

Date 2020-01-23

Author Ottar N. Bjornstad [aut, cre]

Maintainer Ottar N. Bjornstad <`onb1@psu.edu`>

Description Functions, data sets and shiny apps for ``Epidemics: Models and Data in R'' by Ottar N. Bjornstad (ISBN 978-3-319-97487-3) <`https://www.springer.com/gp/book/9783319974866`>. The package contains functions to study the S(E)IR model, spatial and age-structured SIR models; time-series SIR and chain-binomial stochastic models; catalytic disease models; coupled map lattice models of spatial transmission and network models for social spread of infection. The package is also an advanced quantitative companion to the coursera Epidemics Massive Online Open Course <`https://www.coursera.org/learn/epidemics`>.

Depends R (>= 3.3.2), shiny, deSolve, polspline

Suggests ade4, bbmle, fields, forecast, imputeTS, lme4, ncf, nleqslv, nlme, nlts, plotrix, pomp, rootSolve, Rwave, statnet,

License GPL-3

URL `https://github.com/objornstad/epimdr`,
`https://www.springer.com/gp/book/9783319974866`,
`http://ento.psu.edu/directory/onb1`

BugReports `https://github.com/objornstad/epimdr/issues`

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2020-01-25 05:30:02 UTC

Contents

BarabasiAlbert . . . 3
black . . . 4
burnett . . . 4
ccs . . . 5
chainSIR . . . 6
cholera . . . 6
coyne . . . 7
dalziel . . . 7
ebola . . . 8
ferrari . . . 9
filipendula . . . 9
fiv . . . 10
flowField . . . 11
flu . . . 13
gillespie . . . 14
gm . . . 14
gonnet . . . 16
gra . . . 16
Icelandflu . . . 17
integrandpc . . . 17
litter . . . 18
llik.cb . . . 19
llik.pc . . . 19
magono . . . 20
May.app . . . 20
meas . . . 21
mossong . . . 22
NB . . . 22
NetworkSIR . . . 23
niamey . . . 24
niamey_daily . . . 25
orv.app . . . 25
pagiard . . . 26
paili . . . 26
palymes . . . 27
pameasle . . . 27
pertcop . . . 28
peru . . . 28
plot.cm . . . 29
plot.netSIR . . . 30
r0fun . . . 30
rabbit . . . 31
rabies . . . 31
retrospec . . . 32
ringlattice . . . 33
SEIR.app . . . 33

seirmod 34
seirmod2 35
SEIRS.app 35
SH9 36
silene2 37
sim.cb 37
SimTsir 38
SimTsir2 39
SIR.app 39
siragemod 40
sirmod 41
sirwmod 41
sivmod 42
summary.cm 43
summary.netSIR 43
tau 44
TSIR.app 45
TSIRllyap 45
TSIRlyap 46
tydiphtheria 46
tymeasles 47
tyscarlet 48
tywhooping 48
usflu 49
WattsStrogatz 49

Index **51**

`BarabasiAlbert` *Function to generate a Barabasi-Albert network*

Description

Function to generate a Barabasi-Albert network

Usage

```
BarabasiAlbert(N, K)
```

Arguments

`N`	the number of nodes
`K`	the number of neighbors to which each node is connected so degree = 2*K

Value

An object of class CM (contact matrix)

Examples

```
cm3=BarabasiAlbert(200, 4)
```

`black` *Black's measles seroprevalence data.*

Description

Seroprevalence-by-age-bracket for measles in prevaccination New Haven as studied by Black (1959).

Usage

```
black
```

Format

A data frame with 42 rows and 3 variables:

age age-bracket (in years)

mid mid-point of age-bracket (in years)

n number of tests

pos number seropositive

neg number seronegative

f seroprevalence

Source

Black (1959) Measles antibodies in the population of New Haven, Connecticut. Journal of Immunology 83:74-83

`burnett` *Burnett's Parasitoid-Host data.*

Description

Data is of 22 generations of greenhouse white flies (*Trialeurodes vaporariorum*) and its parasitoid, *Encarsia formosa*. Column names are self explanatory.

Usage

```
burnett
```

Format

A data frame with 22 rows and 7 variables:

Generation

NumberofHostsExposed

NumberofHostsParasitized

NumberofHostsUnparasitized

NumberofParasiteEggsLaid

NumberofParasitesSearching

PercentageofHostsParasitized

Source

Burnett, T. A. (1958) Model of host-parasite interaction Proceedings of the 10th International Congress, Entomology, 1958, 2, 679-686

`ccs` *UK measles CCS data.*

Description

The fraction of weeks measles was absent from each of the 954 cities and towns of England and Wales between 1944 and 1965.

Usage

`ccs`

Format

A data frame with 954 rows and 14 variables:

fade3 Average duration of fadeout (of at least 3 weeks of length)

ext Fraction of time when measles was absent

size Median population size

fade Average duration of fadeouts (of a week or longer)

se3 Standard error fade3

se Standard error of fade

n3 The number of fadeouts (of at least 3 weeks of length)

n The number of fadeout of a week or longer

names City/town name

Source

Bjornstad and Grenfell (2008) Hazards, spatial transmission and timing of outbreaks in epidemic metapopulations. Environmental and Ecological Statistics 15: 265-277. doi:10.1007/s10651-007-0059-3.

`chainSIR` *Gradient-function for the chain-SIR model*

Description

Gradient-function for the chain-SIR model

Usage

```
chainSIR(t, logx, params)
```

Arguments

`t`	Implicit argument for time
`logx`	A vector with values for the log-states
`params`	A vector with parameter values for the chain-SIR system

Value

A list of gradients

Examples

```
require(deSolve)
times  = seq(0, 10, by=1/52)
paras2  = c(mu = 1/75, N = 1, beta =  625, gamma = 365/14, u=5)
xstart2 = log(c(S=.06, I=c(0.001, rep(0.0001, paras2["u"]-1)), R = 0.0001))
out = as.data.frame(ode(xstart2, times, chainSIR, paras2))
```

`cholera` *Dacca cholera death data.*

Description

Monthly deaths from cholera in Dacca, East Bengal between 1891 and 1940.

Usage

```
cholera
```

Format

A data frame with 600 rows and 4 variables:

Year Year

Month Month of the year

Dacca Monthly cholera deaths

Population Population size of district

Source

King, A.A., Ionides, E.L., Pascual, M. and Bouma, M. J. (2008) Inapparent infections and cholera dynamics. Nature, 454:877-880. doi.org/10.1038/nature07084.

`coyne` *Gradient-function for Coyne et al's rabies model*

Description

Gradient-function for Coyne et al's rabies model

Usage

```
coyne(t, logx, parms)
```

Arguments

`t`	Implicit argument for time
`logx`	A vector with values for the log-states
`parms`	A vector with parameter values for the dynamical system

Value

A list of gradients for the log system

Examples

```
require(deSolve)
times  = seq(0, 50, by=1/520)
paras  = c(gamma = 0.0397, b = 0.836, a = 1.34, sigma = 7.5,
alpha = 66.36, beta = 33.25, c = 0, rho = 0.8)
start = log(c(X=12.69/2, H1=0.1, H2=0.1, Y = 0.1, I = 0.1))
out = as.data.frame(ode(start, times, coyne, paras))
```

`dalziel` *Measles incidence across 40 US cities*

Description

A dataset of Measles incidence across 40 US cities with relevant demographic data

Usage

```
dalziel
```

Format

A data frame with 44,720 rows and 10 variables:

biweek biweek of the year

cases incidence

year year

loc city name

pop population size

rec susceptible recruits

country country

lon city longitude

lat city latitude

decimalYear time counter

Source

Dalziel et al. 2016. Persistent chaos of measles epidemics in the prevaccination United States caused by a small change in seasonal transmission patterns. PLoS Computational Biology 2016: e1004655. doi.org/10.1371/journal.pcbi.1004655.

`ebola` *Sierra-Leone Ebola 2015 data.*

Description

The daily number of cases of ebola in Sierra Leone during the 2015 epidemic.

Usage

`ebola`

Format

A data frame with 103 rows and 4 variables:

date date

day day

cum_cases cumulative incidence

cases incidence calculated by differencing the cumcases and setting negatives to zero.

Source

http://www.cdc.gov/vhf/ebola/outbreaks/2014-west-africa/cumulative-cases-graphs.html

---

`ferrari` *Ferrari et al. 2005 outbreak data.*

Description

The incidence aggregated by serial interval of a number of outbreaks studied by Ferrari et al. 2005.

Usage

```
ferrari
```

Format

A data frame with 15 rows and 7 variables:

Eboladeaths00 Number of deaths from ebola during the 2000 Uganda outbreak

Ebolacases00 Number of cases of ebola during the 2000 Uganda outbreak

Ebolacases95 Number of cases of ebola during the 1995 DRC outbreak

FMDfarms Number of farms infected with FMD during the 2000-01 UK outbreak

HogCholera Number of cases of swine fever in pigs in the 1997-98 outbreak in the Netherlands

SarsHk Number of cases of SARS in Hong Kong during the 2003 outbreak

SarsSing Number of cases of SARS in Singapore during the 2003 outbreak

Source

Ferrari et al. (2005) Estimation and inference of R-0 of an infectious pathogen by a removal method. Mathematical Biosciences 198: 14-26. doi.org/10.1016/j.mbs.2005.08.002.

`filipendula` *Filipendula rust data.*

Description

Rust infection status of 162 populations of *Filipendula ulmaria* in a Swedish Island archipelago

Usage

```
filipendula
```

Format

A data frame with 162 rows and 4 variables:

y94 infection status in 1994

y95 infection status in 1995

X X coordinate

Y Y coordinate

Source

Smith et al. 2003. Epidemiological patterns at multiple spatial scales: an 11-year study of a Triphragmium ulmariae – Filipendula ulmaria metapopulation. Journal of Ecology, 91(5), pp.890-903. doi.org/10.1046/j.1365-2745.2003.00811.x.

`fiv` *FIV infection in cats.*

Description

Immunological measures on cats infected with different strains of FIV

Usage

```
fiv
```

Format

A data frame with 238 rows and 18 variables:

Id	Individual identifier
CD4	CD4 cell count
CD8B	CD8B cell count
CD25	CD25 cell count
FAS_L	FAS ligand
FAS	FAS
IFNg	Interferon gamma
IL_10	Interleukin 10
IL_12	Interleukin 12
IL_4	Interleukin 4
lymphocyte	lymphocyte count
neutrophils	neutrophil count
TNF_a	Tumor necrosis factor
provirus	provirus count
viremia	viremia
Day	day
No	unique identifier
Treatment	Experimental treatment

Source

Roy et al. 2009. Multivariate statistical analyses demonstrate unique host immune responses to single and dual lentiviral infection. PloS one 4, e7359. doi.org/10.1371/journal.pone.0007359.

`flowField` *Flowfield*

Description

Plots the flow or velocity field for a one- or two-dimensional autonomous ODE system.

Usage

```
flowField(deriv, xlim, ylim, parameters = NULL, system = "two.dim",
  points = 21, col = "gray", arrow.type = "equal",
  arrow.head = 0.05, frac = 1, add = TRUE, xlab = if (system ==
  "two.dim") state.names[1] else "t", ylab = if (system == "two.dim")
  state.names[2] else state.names[1], ...)
```

Arguments

`deriv`	A function computing the derivative at a point for the ODE system to be analysed. Discussion of the required format of these functions can be found in the package vignette, or in the help file for the function `ode`.
`xlim`	In the case of a two-dimensional system, this sets the limits of the first dependent variable in which gradient reflecting line segments should be plotted. In the case of a one-dimensional system, this sets the limits of the independent variable in which these line segments should be plotted. Should be a `numeric` `vector` of `length` two.
`ylim`	In the case of a two-dimensional system this sets the limits of the second dependent variable in which gradient reflecting line segments should be plotted. In the case of a one-dimensional system, this sets the limits of the dependent variable in which these line segments should be plotted. Should be a `numeric` `vector` of `length` two.
`parameters`	Parameters of the ODE system, to be passed to `deriv`. Supplied as a `numeric` `vector`; the order of the parameters can be found from the `deriv` file. Defaults to `NULL`.
`system`	Set to either `"one.dim"` or `"two.dim"` to indicate the type of system being analysed. Defaults to `"two.dim"`.
`points`	Sets the density of the line segments to be plotted; `points` segments will be plotted in the x and y directions. Fine tuning here, by shifting `points` up and down, allows for the creation of more aesthetically pleasing plots. Defaults to `11`.
`col`	Sets the colour of the plotted line segments. Should be a `character` `vector` of `length` one. Will be reset accordingly if it is of the wrong `length`. Defaults to `"gray"`.
`arrow.type`	Sets the type of line segments plotted. If set to `"proportional"` the `length` of the line segments reflects the magnitude of the derivative. If set to `"equal"` the line segments take equal lengths, simply reflecting the gradient of the derivative(s). Defaults to `"equal"`.

`arrow.head`	Sets the length of the arrow heads. Passed to `arrows`. Defaults to `0.05`.
`frac`	Sets the fraction of the theoretical maximum length line segments can take without overlapping, that they can actually attain. In practice, `frac` can be set to greater than 1 without line segments overlapping. Fine tuning here assists the creation of aesthetically pleasing plots. Defaults to `1`.
`add`	Logical. If `TRUE`, the flow field is added to an existing plot. If `FALSE`, a new plot is created. Defaults to `TRUE`.
`xlab`	Label for the x-axis of the resulting plot.
`ylab`	Label for the y-axis of the resulting plot.
`...`	Additional arguments to be passed to either `plot` or `arrows`.

Value

Returns a `list` with the following components (the exact make up is dependent on the value of `system`):

`add`	As per input.
`arrow.head`	As per input.
`arrow.type`	As per input.
`col`	As per input, but with possible editing if a `character` `vector` of the wrong `length` was supplied.
`deriv`	As per input.
`dx`	A `numeric` `matrix`. In the case of a two-dimensional system, the values of the derivative of the first dependent derivative at all evaluated points.
`dy`	A `numeric` `matrix`. In the case of a two-dimensional system, the values of the derivative of the second dependent variable at all evaluated points. In the case of a one-dimensional system, the values of the derivative of the dependent variable at all evaluated points.
`frac`	As per input.
`parameters`	As per input.
`points`	As per input.
`system`	As per input.
`x`	A `numeric` `vector`. In the case of a two-dimensional system, the values of the first dependent variable at which the derivatives were computed. In the case of a one-dimensional system, the values of the independent variable at which the derivatives were computed.
`xlab`	As per input.
`xlim`	As per input.
`y`	A `numeric` `vector`. In the case of a two-dimensional system, the values of the second dependent variable at which the derivatives were computed. In the case of a one-dimensional system, the values of the dependent variable at which the derivatives were computed.
`ylab`	As per input.
`ylim`	As per input.

Author(s)

Michael J Grayling

See Also

`arrows`, `plot`

Examples

```
#See archived phaseR package for examples
```

`flu` *Boarding school influenza data.*

Description

The daily number of children confined to bed in a boarding school in North England during an outbreak in 1978 of the reemerging A/H1N1 strain. The school had 763 boys of which 512 boys were confined to bed sometime during the outbreak.

Usage

```
flu
```

Format

A data frame with 14 rows and 2 variables:

day day since beginning of outbreak

cases number of sick children

Source

Anonymous (1978) EPIDEMIOLOGY: Influenza in a boarding school. British Medical Journal, 4 March 1978 p.587.

`gillespie` *Gillespie exact algorithm*

Description

Function simulating a dynamical system using the Gillespie exact algorithm

Usage

```
gillespie(rateqs, eventmatrix, parameters, initialvals, numevents)
```

Arguments

`rateqs`	a list with rate equations
`eventmatrix`	a matrix of changes in state variables associated with each event
`parameters`	a vector of parameter values
`initialvals`	a vector of initial values for the states
`numevents`	number of events to be simulated

Value

A data frame with simulated time series

Examples

```
rlist=c(quote(mu * (S+I+R)), quote(mu * S), quote(beta * S * I /(S+I+R)),
 quote(mu * I), quote(gamma * I), quote(mu*R))
emat=matrix(c(1,0,0,-1,0,0,-1,1,0,0,-1,0,0,-1,1,0,0,-1),ncol=3, byrow=TRUE)
paras  = c(mu = 1, beta =  1000, gamma = 365/20)
inits = c(S=100, I=2, R=0)
sim=gillespie(rlist, emat, paras, inits, 100)
```

`gm` *Defoliated by gypsy moth each in northeast US 1975-2002.*

Description

A dataset containing the fraction of forest defoliated by the gypsy moth in 20km x 20km pixels across northeast US in each year between 1975 and 2002.

Usage

```
gm
```

Format

A data frame with 1086 rows and 30 variables:

UTMX	UTM x-coordinates
UTMY	UTM y-coordinates
d1975	Defolitation in 1975
d1976	Defolitation in 1976
d1977	Defolitation in 1977
d1978	Defolitation in 1978
d1979	Defolitation in 1979
d1980	Defolitation in 1980
d1981	Defolitation in 1981
d1982	Defolitation in 1982
d1983	Defolitation in 1983
d1984	Defolitation in 1984
d1985	Defolitation in 1985
d1986	Defolitation in 1986
d1987	Defolitation in 1987
d1988	Defolitation in 1988
d1989	Defolitation in 1989
d1990	Defolitation in 1990
d1991	Defolitation in 1991
d1992	Defolitation in 1992
d1993	Defolitation in 1993
d1994	Defolitation in 1994
d1995	Defolitation in 1995
d1996	Defolitation in 1996
d1997	Defolitation in 1997
d1998	Defolitation in 1998
d1999	Defolitation in 1999
d2000	Defolitation in 2000
d2001	Defolitation in 2001
d2002	Defolitation in 2002

Source

Bjornstad, O. N., Robinet, C., & Liebhold, A. M. (2010). Geographic variation in North American gypsy moth cycles: subharmonics, generalist predators, and spatial coupling. Ecology, 91(1), 106-118. doi.org/10.1890/08-1246.1.

`gonnet` *De et al. 2004 gonorrhea contact matrix*

Description

The directed contact network from De et al. (2004) contact-tracing of the spread of gonorrhea across asexual network in Alberta canada

Usage

```
gonnet
```

Format

A matrix with 89 rows and 89 columns:

gonet a matrix of directional contacts of disease spread

Source

De et al (2004). Sexual network analysis of a gonorrhea outbreak. Sexually transmitted infections 80: 280-285. doi.org/10.1136/sti.2003.007187.

`gra` *Euthamia graminifolia rust data.*

Description

Data on a fungal pathogen of the aster Euthamia graminifolia collected by Jennifer Keslow.

Usage

```
gra
```

Format

A data frame with 360 rows and 8 variables:

block the block
row row
plot plot within block
xloc x coordinates
yloc y coordinate
comp plot composition
water treatment: dry or wet
score the rust score

`Icelandflu` *Monthly incidence of influenza-like illness in Iceland between 1980 and 2009.*

Description

A dataset containing the monthly ILI incidence in Iceland between 1980 and 2009.

Usage

```
Icelandflu
```

Format

A data frame with 360 rows and 3 variables:

month the month

year the year

ili ILI incidence

Source

Bjornstad ON, Viboud C. Timing and periodicity of influenza epidemics. Proceedings of the National Academy of Sciences. 2016 Nov 15;113(46):12899-901. doi.org/10.1073/pnas.1616052113.

`integrandpc` *Auxillary function used by llik.pc*

Description

Auxillary function used by llik.pc

Usage

```
integrandpc(a, up, foi)
```

Arguments

`a`	a vector with the ages
`up`	a vector with upper age-bracket cut-offs
`foi`	a vector with FoI

Value

A vector with FoIs matched to data

See Also

llik.pc

`litter` *Bordetella bronchiseptica in rabbit kittens.*

Description

Data on Bordetella bronchiseptica in rabbit kittens in a breeding facility.

Usage

```
litter
```

Format

A data frame with 494 rows and 8 variables:

Facility	breeding facility
sick	infection status
Date	date sampled
Animal.code	animal identifier
msick	dams infection status
Litter	litter identifier
CFU	bacterial count
Description	unique litter identifier

Source

Long et al (2010) Identifying the Age Cohort Responsible for Transmission in a Natural Outbreak of Bordetella bronchiseptica. PLoS Pathogens 6(12): e1001224. doi:10.1371/journal.ppat.1001224.

`llik.cb` *Negative log-likelihood function for the chain-binomial model*

Description

Negative log-likelihood function for the chain-binomial model

Usage

```
llik.cb(S0, beta, I)
```

Arguments

`S0`	a scalar with value for S0
`beta`	a scalar with value for beta
`I`	a vector incidence aggregated at serial interval

Value

the negative log-likelhood for the model

Examples

```
twoweek=rep(1:15, each=2)
niamey_cases1=sapply(split(niamey$cases_1[1:30], twoweek), sum)
llik.cb(S0=6500, beta=23, I=niamey_cases1)
```

`llik.pc` *Function to estimate parameters for the picewise-constant catalytic model*

Description

This function uses binomial likelihoods to estimate the picewise-constant FoI model from age-incidence data

Usage

```
llik.pc(par, age, num, denom, up)
```

Arguments

`par`	a vector with initial guesses
`age`	a vector with the ages
`num`	a vector with number infected by age
`denom`	a vector with number tested by age
`up`	a vector with upper age-bracket cut-offs

Value

The negative log-likelihood for a candidate piecewise constant catalytic model

Examples

```
x=c(1,4,8,12,18,24)
para=rep(.1,length(x))
## Not run: optim(par=log(para),fn=loglikpc, age=rabbit$a, num=rabbit$inf, denom=rabbit$n, up=x)
```

`magono` *Massachusetts gonorrhea data.*

Description

Weekly cases of gonorrhea in Massachusetts between 2006 and 2015.

Usage

```
magono
```

Format

A data frame with 422 rows and 4 variables:

number Weekly case reports

year Year

week Week of the year

time Time in fractions of year

Source

`https://www.tycho.pitt.edu`

`May.app` *Launch a shiny-app simulating May's Parasitoid-host Model model*

Description

Launch a shiny-app simulating May's Parasitoid-host Model model

Usage

```
May.app
```

Format

An object of class `shiny.appobj` of length 5.

Details

Launch app for details

Examples

```
## Not run: May.app
```

`meas` *Bi-weekly measles incidence in London from 1944-65.*

Description

A dataset containing the biweekly incidence of measles in London from 1944 to 1965

Usage

```
meas
```

Format

A data frame with 546 rows and 5 variables:

year year

week week of the year

time time

London incidence

B Biweekly births

Details

Birth numbers are annual, so in the data set, this number is evenly distributed across the 26 bi-weeks of each year.

Source

Bjornstad et al. (2002) Endemic and epidemic dynamics of measles: Estimating transmission rates and their scaling using a time series SIR model. Ecological Monographs 72: 169-184. doi.org/10.2307/3100023.

`mossong` *POLYMOD contact-rate data by Age.*

Description

Age-specific contact rates from the diary study by Mossong et al. 2008.

Usage

```
mossong
```

Format

A data frame with 900 rows and 3 variables:

contactor end of age-bracket (in years) of contactor group

contactee end of age-bracket (in years) of contactee group

contact.rate average contact rate

Source

Mossong et al. 2008 Social contacts and mixing patterns relevant to the spread of infectious diseases PLoS Med, Public Library of Science 5:e74. doi.org/10.1371/journal.pmed.0050074.

`NB` *The Nicholson-Bailey model*

Description

Function to simulate the Nicholson-Bailey Parasitoid-host model

Usage

```
NB(R, a, T = 100, H0 = 10, P0 = 1)
```

Arguments

`R`	the host reproductive rate
`a`	the parasitoid search efficiency
`T`	the length of simulation (number of time-steps)
`H0`	initial host numbers
`P0`	initial parasitoid numbers

Value

A list of simulated Host and Parasitoid numbers

Examples

```
sim= NB(R=1.1,a=0.1)
```

`NetworkSIR` *Function to simulate an epidemic on a network*

Description

Function to simulate a stochastic (discrete time) Reed-Frost SIR model on a social network

Usage

```
NetworkSIR(CM, tau, gamma)
```

Arguments

`CM`	a contact matrix
`tau`	the transmission probability
`gamma`	the recovery probability

Value

An object of class netSIR with infectious status for each node through time

Examples

```
cm1=BarabasiAlbert(N=200,K=2)
sim1=NetworkSIR(cm1,.3,0.1)
summary(sim1)
## Not run: plot(sim1)
```

`niamey` *Weekly measles incidence from 2003-04 in Niamey, Niger.*

Description

A dataset containing the weekly incidence of measles in Niamey, Niger during the 2003-04 outbreak

Usage

```
niamey
```

Format

A data frame with 31 rows and 13 variables:

absweek	week since beginning of outbreak
week	week of the year
tot_cases	weekly incidence for the whole city
tot_mort	weekly deaths for the whole city
lethality	weekly case fatality rate
tot_attack	weekly attack rates for the whole city
cases_1	weekly incidence for district 1
attack_1	weekly attack rates for district 1
cases_2	weekly incidence for district 2
attack_2	weekly attack rates for district 2
cases_3	weekly incidence for district 3
attack_3	weekly attack rates for district 3
cum_cases	weekly cumulative incidence for the whole city

Source

Grais et al (2008) Time is of the essence: exploring a measles outbreak response vaccination in Niamey, Niger. Journal of the Royal Society Interface 5: 67-74. https://doi.org/10.1098/rsif.2007.1038.

`niamey_daily` *Day of appearance of each measles case from 2003-04 outbreak in Niamey, Niger.*

Description

A dataset containing the day of appearance of each measles case in Niamey, Niger during the 2003-04 outbreak.

Usage

```
niamey_daily
```

Format

A data frame with 10,937 rows and 1 variables:

day the day of appearance of each case since day of outbreak

Source

Grais et al. (2008) Time is of the essence: exploring a measles outbreak response vaccination in Niamey, Niger. Journal of the Royal Society Interface 5: 67-74. doi.org/10.1098/rsif.2007.1038.

`orv.app` *Launch a shiny-app to study outbreak-response vaccination campaigns*

Description

Launch a shiny-app to study outbreak-response vaccination campaigns

Usage

```
orv.app
```

Format

An object of class `shiny.appobj` of length 5.

Details

Launch app for details

Examples

```
## Not run: orv.app
```

`pagiard` *Weekly incidence of giardia in Pennsylvania between 2006 and 2014.*

Description

A dataset containing the weekly incidence of giardia in Pennsylvania between 2006 and 2014.

Usage

```
pagiard
```

Format

A data frame with 448 rows and 3 variables:

PENNSYLVANIA weekly incidence

YEAR the year

WEEK the week

Source

https://www.tycho.pitt.edu

`paili` *Weekly deaths from Influenza-like illness in Pennsylvania between 1972 and 1998.*

Description

A dataset containing the weekly ILI related deaths in Pennsylvania between 1972 and 1998.

Usage

```
paili
```

Format

A data frame with 1404 rows and 3 variables:

PENNSYLVANIA weekly deaths

YEAR the year

WEEK the week

Source

https://www.tycho.pitt.edu

`palymes` *Weekly incidence of Lymes disease in Pennsylvania between 2006 and 2014.*

Description

A dataset containing the weekly incidence of Lymes disease in Pennsylvania between 2006 and 2014.

Usage

```
palymes
```

Format

A data frame with 448 rows and 3 variables:

PENNSYLVANIA weekly incidence

YEAR the year

WEEK the week

Source

https://www.tycho.pitt.edu

`pameasle` *Weekly incidence of measles in Pennsylvania between 1928 and 1969.*

Description

A dataset containing the weekly incidence of measles in Pennsylvania between 2006 and 2014.

Usage

```
pameasle
```

Format

A data frame with 448 rows and 3 variables:

PENNSYLVANIA weekly incidence

YEAR the year

WEEK the week

Source

https://www.tycho.pitt.edu

`pertcop` *Weekly whooping cough incidence from 1900-1937 in Copenhagen, Denmark.*

Description

A dataset containing the weekly incidence of whooping cough from Copenhagen, Denmark between January 1900 and December 1937

Usage

```
pertcop
```

Format

A data frame with 1982 rows and 9 variables:

date date

births births

day day of month

month month of year

year year

cases weekly incidence

deaths weekly deaths

popsize weekly population size interpolated from census data

Source

Lavine et al. 2013. Immune boosting explains regime- shifts in prevaccine-era pertussis dynamics. PLoS One, 8(8):e72086. doi:10.1371/journal.pone.0072086.

`peru` *Rubella in Peru data.*

Description

Rubella incidence by age as studied by Metcalf et al (2011).

Usage

```
peru
```

Format

A data frame with 95 rows and 2 variables:

age end of age-bracket (in years)

cumulative cumulative number of rubella cases

incidence number of rubella cases

n total cases

Source

Metcalf et al (2011) Rubella metapopulation dynamics and importance of spatial coupling to the risk of congenital rubella syndrome in Peru. Journal of the Royal Society Interface 8: 369-376. doi:10.1371/journal.pone.0072086.

`plot.cm` *Function to plot an object of class CM*

Description

Function to plot an object of class CM

Usage

```
## S3 method for class 'cm'
plot(x, ...)
```

Arguments

`x`	an object of class cm
`...`	other arguments

Value

A plot of the contract matrix

Examples

```
cm=ringlattice(N=20,K=4)
## Not run: plot(cm)
```

`plot.netSIR` *Function to plot a netSIR object*

Description

Function to plot a netSIR object

Usage

```
## S3 method for class 'netSIR'
plot(x, ...)
```

Arguments

`x`	an object of class netSIR
`...`	other arguments

See Also

`netSIR`

`r0fun` *Function to calculate R0 from a contact matrix*

Description

Function to calculate R0 from a contact matrix

Usage

```
r0fun(CM, tau, gamma)
```

Arguments

`CM`	an object of class CM
`tau`	= probability of infection across an edge
`gamma`	= probability of removal per time step

Value

the R0

Examples

```
cm1=BarabasiAlbert(N=200,K=2)
r0fun(cm1, 0.3, 0.1)
```

`rabbit` *Rabbit* Bordetella brochiseptica *data.*

Description

Rabbits infected by *B. brochiseptica* by age as studied by Long et al (2010).

Usage

```
rabbit
```

Format

A data frame with 42 rows and 3 variables:

a end of age-bracket (in months)

n number of rabbits tested

inf number of rabbits infected with the bacterium

Source

Long et al (2010) Identifying the Age Cohort Responsible for Transmission in a Natural Outbreak of Bordetella bronchiseptica. PLoS Pathogens 6(12): e1001224. doi:10.1371/journal.ppat.1001224.

`rabies` *Raccoon rabies data.*

Description

Data is the average monthly number of reported cases of rabid raccoons across all counties within each of 11 east coast US states the time line is from the first reported case in each state (starting in late 1970s for West Virginia).

Usage

```
rabies
```

Format

A data frame with 208 rows and 12 variables:

Month Month since rabies appearance in the state

CT Connecticut

DE Delaware

MD Maryland

MA Massachusetts

NJ New Jersey

NY New York

NC North Carolina

PA Pennsylvania

RI Rhode Island

VA Virginia

WV West Virginia

Source

Childs et al. 2000. Predicting the local dynamics of epizootic rabies among raccoons in the United States Proceedings of the National Academy of Sciences 97:13666-13671. doi.org/10.1073/pnas.240326697.

`retrospec` *Function to predict efficacy of outbreak-response vaccination campaign*

Description

Function to predict efficacy of outbreak-response vaccination campaign

Usage

```
retrospec(R, day, vaccine_efficacy, target_vaccination,
  intervention_length, mtime, LP = 7, IP = 7, N = 10000)
```

Arguments

`R`	reproductive ratio
`day`	first day of ORV campaign
`vaccine_efficacy`	Vaccine efficacy
`target_vaccination`	fraction of population vaccinated during ORV campaign
`intervention_length`	duration of ORV campaign
`mtime`	length of simulation
`LP`	length of latent period
`IP`	length of infectious period
`N`	initial susceptible population size

Value

A list of gradients

Examples

```
red1=retrospec(R=1.8, 161, vaccine_efficacy=0.85, target_vaccination=0.5,
 intervention_length=10, mtime=250, LP=8, IP=5, N=16000)
1-red1$redn
```

`ringlattice` *Function to generate a ring lattice*

Description

Function to generate a ring lattice

Usage

```
ringlattice(N, K)
```

Arguments

`N`	the number of nodes
`K`	the number of neighbors to which each node is connected so degree = 2xK

Value

An object of class CM (contact matrix)

Examples

```
cm=ringlattice(N=20,K=4)
```

`SEIR.app` *Launch a shiny-app simulating the seasonal SEIR model*

Description

Launch a shiny-app simulating the seasonal SEIR model

Usage

```
SEIR.app
```

Format

An object of class `shiny.appobj` of length 5.

Details

Launch app for details

Examples

```
## Not run: SEIR.app
```

`seirmod` *Gradient-function for the SEIR model*

Description

Gradient-function for the SEIR model

Usage

```
seirmod(t, y, parms)
```

Arguments

`t`	Implicit argument for time
`y`	A vector with values for the states
`parms`	A vector with parameter values for the SEIR system

Value

A list of gradients

Examples

```
require(deSolve)
times  = seq(0, 10, by=1/120)
paras  = c(mu = 1/50, N = 1, beta =  1000, sigma = 365/8, gamma = 365/5)
start = c(S=0.06, E=0, I=0.001, R = 0.939)
out=ode(y=start, times=times, func=seirmod, parms=paras)
```

`seirmod2` *Gradient-function for the forced SEIR model*

Description

Gradient-function for the forced SEIR model

Usage

```
seirmod2(t, y, parms)
```

Arguments

`t`	Implicit argument for time
`y`	A vector with values for the states
`parms`	A vector with parameter values for the SIR system

Value

A list of gradients

Examples

```
require(deSolve)
times  = seq(0, 10, by=1/120)
paras  = c(mu = 1/50, N = 1, beta0 = 1000, beta1 = 0.2, sigma = 365/8, gamma = 365/5)
start = c(S=0.06, E=0, I=0.001, R = 0.939)
out=ode(y=start, times=times, func=seirmod2, parms=paras)
```

`SEIRS.app` *Launch a shiny-app simulating the SEIRS model*

Description

Launch a shiny-app simulating the SEIRS model

Usage

```
SEIRS.app
```

Format

An object of class `shiny.appobj` of length 5.

Details

Launch app for details

Examples

```
## Not run: SEIRS.app
```

`SH9` *Daily measures of malaria infected mice.*

Description

Daily data on laboratory mice infected with various strains of *Plasmodium chaudaudi*

Usage

```
SH9
```

Format

A data frame with 1300 rows and 11 variables:

Line	line number
Day	day of infection
Box	Cage number
Mouse	Mouse identifier
Treatment	Plasmodium strain
Ind2	Unique mouse identifier
Weight	Mouse weight
Glucose	Blood glucose level
RBC	Red blood cell count
Sample	Sample number
Para	Parasite count

Source

Sylvie Huijben

`silene2` *Antler smut on wild campion.*

Description

Data on a fungal pathogen of the wild campion collected by Janis Antonovics

Usage

```
silene2
```

Format

A data frame with 876 rows and 5 variables:

X	road segment number
lat	latitude
long	longitude
hmean	number of healthy plants
dmean	number of diseased plants

Source

Antonovics, J. 2004. Long-term study of a plant-pathogen metapopulation. In: Hanski, Ilkka, and Oscar E. Gaggiotti. Ecology, genetics, and evolution of metapopulations. Academic Press. doi.org/10.1371/journal.pone.0007359.

`sim.cb` *Function to simulate the chain-binomial model*

Description

Function to simulate the chain-binomial model

Usage

```
sim.cb(S0, beta)
```

Arguments

`S0`	a scalar with value for S0
`beta`	a scalar with value for beta

Value

A data-frame with time series of susceptibles and infecteds

Examples

```
sim=sim.cb(S0=6500, beta=23)
```

`SimTsir` *Function to simulate the stochastic TSIR*

Description

Function to simulate the stochastic TSIR assuming stochasticity in transmission and a Poisson birth-death process

Usage

```
SimTsir(alpha = 0.97, B = 2300, beta = 25, sdbeta = 0, S0 = 0.06,
  I0 = 180, IT = 520, N = 3300000)
```

Arguments

`alpha`	the exponent on I
`B`	the birth rate
`beta`	the transmission rate
`sdbeta`	the standard deviation on beta
`S0`	the initial susceptible fraction
`I0`	the initial number of infecteds
`IT`	the length of simulation
`N`	the population size

Value

A list with time series of simulated infected and susceptible hosts

Examples

```
out = SimTsir()
```

`SimTsir2` *Function to simulate the seasonally-forced TSIR*

Description

Function to simulate the stochastic TSIR assuming stochasticity in transmission and a Poisson birth-death process

Usage

```
SimTsir2(beta, alpha, B, N, inits = list(Snull = 0, Inull = 0),
  type = "det")
```

Arguments

`beta`	the seasonal transmission coefficients
`alpha`	the exponent on I
`B`	a vector of Births (the length of which determines the length of the simulation)
`N`	the population size
`inits`	a list containing initial S and I
`type`	an argument "det" or "stoc" that determines whether a deterministic or stochastic simulation is done

Value

A list with time series of simulated infected and susceptible hosts

Examples

```
## Not run: see chapter 8 in book
```

`SIR.app` *Launch a shiny-app simulating the SIR model*

Description

Launch a shiny-app simulating the SIR model

Usage

```
SIR.app
```

Format

An object of class `shiny.appobj` of length 5.

Details

Launch app for details

Examples

```
## Not run: SIR.app
```

`siragemod` *Gradient-function for the age-structured SIR model with possibly heterogeneous mixing*

Description

Gradient-function for the age-structured SIR model with possibly heterogeneous mixing

Usage

```
siragemod(t, logx, parms)
```

Arguments

`t`	Implicit argument for time
`logx`	A vector with log-values for the log-states
`parms`	A vector with parameter values for the age-structured SIR system

Value

A list of gradients

Examples

```
a=rep(1,4)
n=length(a)
betaM=matrix(1, ncol=4, nrow=4)
pars =list(N=1, gamma=365/14, mu=0.02, sigma=0.2, beta=500, betaM=betaM,p=rep(0,4), a=a)
xstart<-log(c(S=rep(0.099/n,n), I=rep(0.001/n,n), R=rep(0.9/n,n)))
times=seq(0,10,by=14/365)
out=as.data.frame(ode(xstart, times=times, func=siragemod, parms=pars))
```

`sirmod` *Gradient-function for the SIR model*

Description

Gradient-function for the SIR model

Usage

```
sirmod(t, y, parms)
```

Arguments

`t`	Implicit argument for time
`y`	A vector with values for the states
`parms`	A vector with parameter values for the SIR system

Value

A list of gradients

Examples

```
require(deSolve)
times  = seq(0, 26, by=1/10)
paras  = c(mu = 0, N = 1, beta =  2, gamma = 1/2)
start = c(S=0.999, I=0.001, R = 0)
out=ode(y=start, times=times, func=sirmod, parms=paras)
```

`sirwmod` *Gradient-function for the SIRWS model*

Description

Gradient-function for the SIRWS model

Usage

```
sirwmod(t, logy, parms)
```

Arguments

`t`	Implicit argument for time
`logy`	A vector with values for the log(states)
`parms`	A vector with parameter values for the SIRWS system

Value

A list of gradients (in log-coordinates)

Examples

```
require(deSolve)
times  = seq(0, 26, by=1/10)
paras  = c(mu = 1/70, p=0.2, N = 1, beta = 200, omega = 1/10, gamma = 17, kappa=30)
start = log(c(S=0.06, I=0.01, R=0.92, W = 0.01))
out = as.data.frame(ode(start, times, sirwmod, paras))
```

`sivmod` *Gradient-function for the SIR model with outbreak-response vaccination*

Description

Gradient-function for the SIR model with outbreak-response vaccination

Usage

```
sivmod(t, x, parms)
```

Arguments

`t`	Implicit argument for time
`x`	A vector with values for the states
`parms`	A vector with parameter values for the SIR system

Value

A list of gradients

See Also

`retrospec`

`summary.cm` *Function to calculate the degree distribution for an object of class CM*

Description

Function to calculate the degree distribution for an object of class CM

Usage

```
## S3 method for class 'cm'
summary(object, plot = FALSE, ...)
```

Arguments

`object`	an object of class cm
`plot`	if TRUE a bar plot of the degree distribution is produced
`...`	other arguments

Value

A plot of the contract matrix

Examples

```
cm=WattsStrogatz(N=20, K=4, Prw=.3)
summary(cm)
```

`summary.netSIR` *Function to summarize a netSIR object*

Description

Function to summarize a netSIR object

Usage

```
## S3 method for class 'netSIR'
summary(object, ...)
```

Arguments

`object`	an object of class netSIR
`...`	other arguments

Value

A data-frame with the time series of susceptible, infected and recovered individuals

See Also

`netSIR`

`tau` *Gillespie tau-leap algorithm*

Description

Function simulating a dynamical system using the Gillespie tau-leap approximation

Usage

```
tau(rateqs, eventmatrix, parameters, initialvals, deltaT, endT)
```

Arguments

`rateqs`	a list with rate equations
`eventmatrix`	a matrix of changes in state variables associated with each event
`parameters`	a vector of parameter values
`initialvals`	a vector of initial values for the states
`deltaT`	the tau-leap time interval
`endT`	the time length of simulation

Value

A data frame with simulated time series

Examples

```
rlist2=c(quote(mu * (S+E+I+R)), quote(mu * S), quote(beta * S * I/(S+E+I+R)),
 quote(mu*E), quote(sigma * E), quote(mu * I), quote(gamma * I), quote(mu*R))
emat2=matrix(c(1,0,0,0,-1,0,0,0,-1,1,0,0,0,-1,0,0,0,-1,1,0,0,0,-1,0,0,0,-1,1,0,0,0,-1),
ncol=4, byrow=TRUE)
paras  = c(mu = 1, beta =  1000, sigma = 365/8, gamma = 365/5)
inits = c(S=999, E=0, I=1, R = 0)
sim2=tau(rlist2, emat2, paras, inits, 1/365, 1)
```

`TSIR.app` *Launch a shiny-app simulating TSIR model*

Description

Launch a shiny-app simulating TSIR model

Usage

```
TSIR.app
```

Format

An object of class `shiny.appobj` of length 5.

Details

Launch app for details

Examples

```
## Not run: TSIR.app
```

`TSIRllyap` *Function to calculate the local Lyapunov exponents for the TSIR*

Description

Function to calculate the local Lyapunov exponents from an object of class `lyap`.

Usage

```
TSIRllyap(x, m = 1)
```

Arguments

`x`	an object of class `lyap` (normally from a call to `TSIRlyap`)
`m`	number of forward iterations on the attractor

Value

An object of class llyap with the local Lyapunov exponent and S-I data

Examples

```
## Not run: see chapter 10 in book
```

`TSIRlyap` *Function to do Lyapunov exponent calculations from a TSIR simulation*

Description

Function to do Lyapunov exponent calculations from a TSIR simulation

Usage

```
TSIRlyap(I, S, alpha, bt, N)
```

Arguments

`I`	a vector containing the time series of Is
`S`	vector containing the time series of Ss
`alpha`	the exponent on I
`bt`	the seasonal transmission coefficients
`N`	the population size

Value

An object of class lyap with the lyapunov exponent, values for the Jacobians, parameters and data

Examples

```
## Not run: see chapter 10 in book
```

`tydiphtheria` *Weekly incidence of diphtheria in Philadelphia between 1914 and 1947.*

Description

A dataset containing the weekly incidence incidence of diphtheria in Philadelphia between 1914 and 1947.

Usage

```
tydiphtheria
```

Format

A data frame with 1774 rows and 4 variables:

YEAR the year

WEEK the week

PHILADELPHIA weekly diphtheria incidence

TIME the time counter

Source

https://www.tycho.pitt.edu

`tymeasles` *Weekly incidence of measles in Philadelphia between 1914 and 1947.*

Description

A dataset containing the weekly incidence incidence of measles in Philadelphia between 1914 and 1947.

Usage

```
tymeasles
```

Format

A data frame with 1774 rows and 4 variables:

YEAR the year

WEEK the week

PHILADELPHIA weekly measles incidence

TIME the time counter

Source

https://www.tycho.pitt.edu

`tyscarlet`	*Weekly incidence of scarlet fever in Philadelphia between 1914 and 1947.*

Description

A dataset containing the weekly incidence incidence of scarlet fever in Philadelphia between 1914 and 1947.

Usage

```
tyscarlet
```

Format

A data frame with 1774 rows and 4 variables:

YEAR the year
WEEK the week
PHILADELPHIA weekly scarlet fever incidence
TIME the time counter

Source

https://www.tycho.pitt.edu

`tywhooping`	*Weekly incidence of whooping cough in Philadelphia between 1925 and 1947.*

Description

A dataset containing the weekly incidence incidence of whooping cough in Philadelphia between 1925 and 1947.

Usage

```
tywhooping
```

Format

A data frame with 1200 rows and 5 variables:

YEAR the year
WEEK the week
PHILADELPHIA weekly whooping cough incidence
TIME the time counter
TM observation counter

Source

https://www.tycho.pitt.edu

`usflu` *US 1975/76 ILI data.*

Description

Influenza-like illness data for the lower 48 states and the District of Columbia during the 1975/76 season dominated by A/H3N2/Victoria strain

Usage

```
usflu
```

Format

A data frame with 49 rows and 7 variables:

State State number

Acronym State code

Pop Population size

Latitude Latitude

Longitude Longitude

Start Week of start of epidemic

Peak Week of peak of epidemic

Source

Viboud C, Bjornstad ON, Smith DL, Simonsen L, Miller MA, Grenfell BT (2006) Synchrony, waves, and spatial hierarchies in the spread of influenza. Science 312: 447-451. doi.org/10.1126/science.1125237.

`WattsStrogatz` *Function to generate a Watts-Strogats network*

Description

Function to generate a Watts-Strogats network

Usage

```
WattsStrogatz(N, K, Prw)
```

Arguments

`N`	the number of nodes
`K`	the number of neighbors to which each node is connected so degree = 2*K
`Prw`	the rewiring probability

Value

An object of class CM (contact matrix)

Examples

```
cm2=WattsStrogatz(N=20, K=4, Prw=.3)
```

Index

∗ **datasets**
  - black, 4
  - burnett, 4
  - ccs, 5
  - cholera, 6
  - dalziel, 7
  - ebola, 8
  - ferrari, 9
  - filipendula, 9
  - fiv, 10
  - flu, 13
  - gm, 14
  - gonnet, 16
  - gra, 16
  - Icelandflu, 17
  - litter, 18
  - magono, 20
  - May.app, 20
  - meas, 21
  - mossong, 22
  - niamey, 24
  - niamey_daily, 25
  - orv.app, 25
  - pagiard, 26
  - paili, 26
  - palymes, 27
  - pameasle, 27
  - pertcop, 28
  - peru, 28
  - rabbit, 31
  - rabies, 31
  - SEIR.app, 33
  - SEIRS.app, 35
  - SH9, 36
  - silene2, 37
  - SIR.app, 39
  - TSIR.app, 45
  - tydiphtheria, 46
  - tymeasles, 47
  - tyscarlet, 48
  - tywhooping, 48
  - usflu, 49

arrows, *12*, *13*

BarabasiAlbert, 3
black, 4
burnett, 4

ccs, 5
chainSIR, 6
character, *11*, *12*
cholera, 6
coyne, 7

dalziel, 7

ebola, 8

ferrari, 9
filipendula, 9
fiv, 10
flowField, 11
flu, 13

gillespie, 14
gm, 14
gonnet, 16
gra, 16

Icelandflu, 17
integrandpc, 17

length, *11*, *12*
list, *12*
litter, 18
llik.cb, 19
llik.pc, 19

magono, 20

matrix, *12*
May.app, 20
meas, 21
mossong, 22

NB, 22
netSIR, *30*, *44*
netSIR (NetworkSIR), 23
NetworkSIR, 23
niamey, 24
niamey_daily, 25
numeric, *11, 12*

ode, *11*
orv.app, 25

pagiard, 26
paili, 26
palymes, 27
pameasle, 27
pertcop, 28
peru, 28
plot, *12, 13*
plot.cm, 29
plot.netSIR, 30

r0fun, 30
rabbit, 31
rabies, 31
retrospec, 32, *42*
ringlattice, 33

SEIR.app, 33
seirmod, 34
seirmod2, 35
SEIRS.app, 35
SH9, 36
silene2, 37
sim.cb, 37
SimTsir, 38
SimTsir2, 39
SIR.app, 39
siragemod, 40
sirmod, 41
sirwmod, 41
sivmod, 42
summary.cm, 43
summary.netSIR, 43

tau, 44
TSIR.app, 45
TSIRllyap, 45
TSIRlyap, 46
tydiphtheria, 46
tymeasles, 47
tyscarlet, 48
tywhooping, 48

usflu, 49

vector, *11, 12*

WattsStrogatz, 49